IMPRINT Workshop

IMPRINT Team

Mr. John Lockett
jlockett@arl.army.mil
410-278-5875

Ms. Jody Wojciechowski
jqw@arl.army.mil
410-278-8830

Ms. Celine Richer
cricher@arl.army.mil
410-278-5883

Ms. Charneta Samms
csamms@arl.army.mil
410-278-5877

Ms. Diane Mitchell
diane@arl.army.mil
410-278-5878

Workshop Objectives

You will be able to…

- Develop an advanced analysis using Advanced or Goal-oriented options
- Interpret results to complete analysis
- Demonstrate Advanced Workload techniques
- Demonstrate Goal Matrix programming
- Identify opportunities for use of IMPRINT capabilities

Development of Analysis

OR

D . O . A

Why Modeling?

Many Variables

Concept System

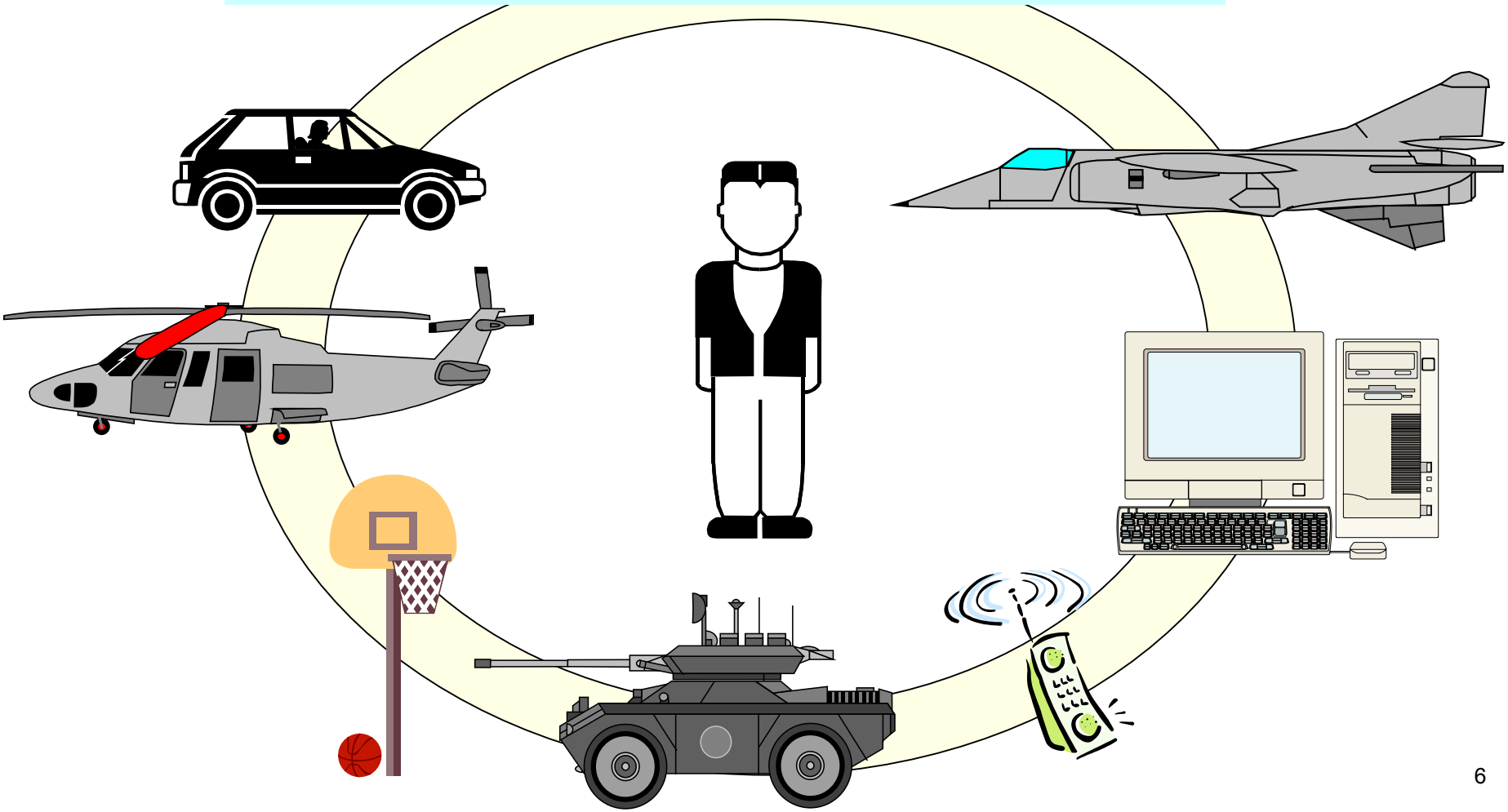
Too Dangerous

Field Study Not Feasible

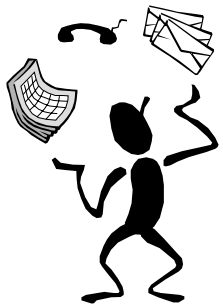
Model – Test – Model

Why Human Performance Modeling?

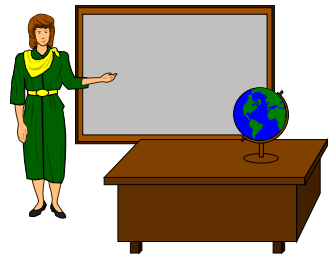
System Performance = f(human performance)



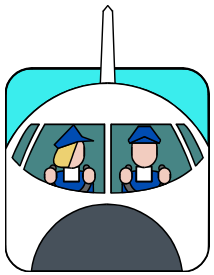
What Does Human Performance Modeling Tell Us?



Is the human overloaded with tasks?



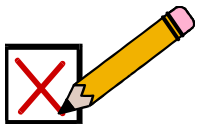
Will training improve human and system performance?



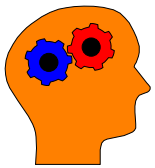
How to allocate tasks between human(s) and automation?

What are the performance tradeoffs with different system designs or levels of operator experience?

Typical Measures



Task time and accuracy



Operator workload level

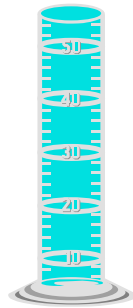
Number of operators required

Impact on System Performance

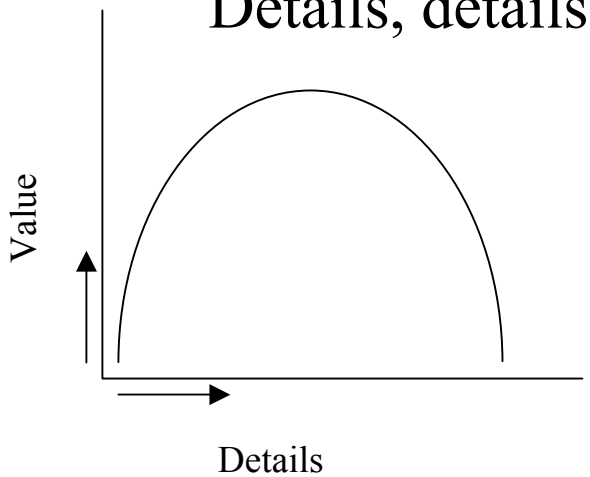
Challenges to Human Performance Modeling

Clear questions

Appropriate measures



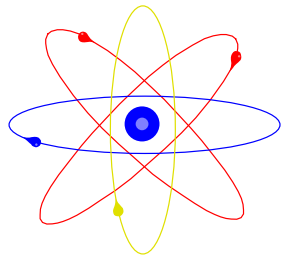
Details, details, details

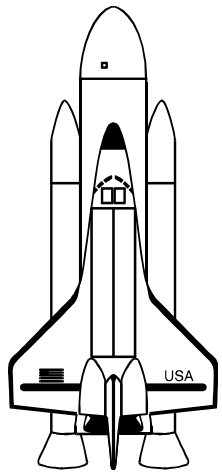


Input data collection

Scientific Method

- Determine the problem - What is your question? Observation!
- Make a hypothesis - What is your prediction?
- Test your hypothesis - What steps and measures are necessary? What tool?
- Analyze your results
- Draw conclusions

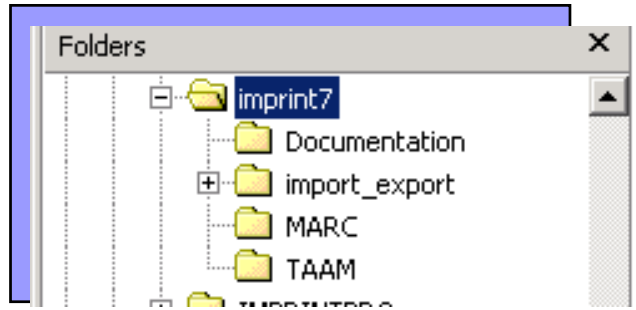




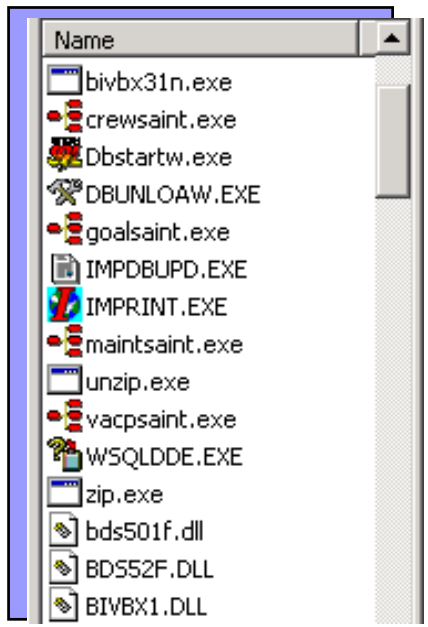
McLaughlin, C.W. and Thompson, M. (1999). Physical Science. Columbus, OH: Glencoe/McGraw-Hill.

IMPRINT Review

The IMPRINT Directory



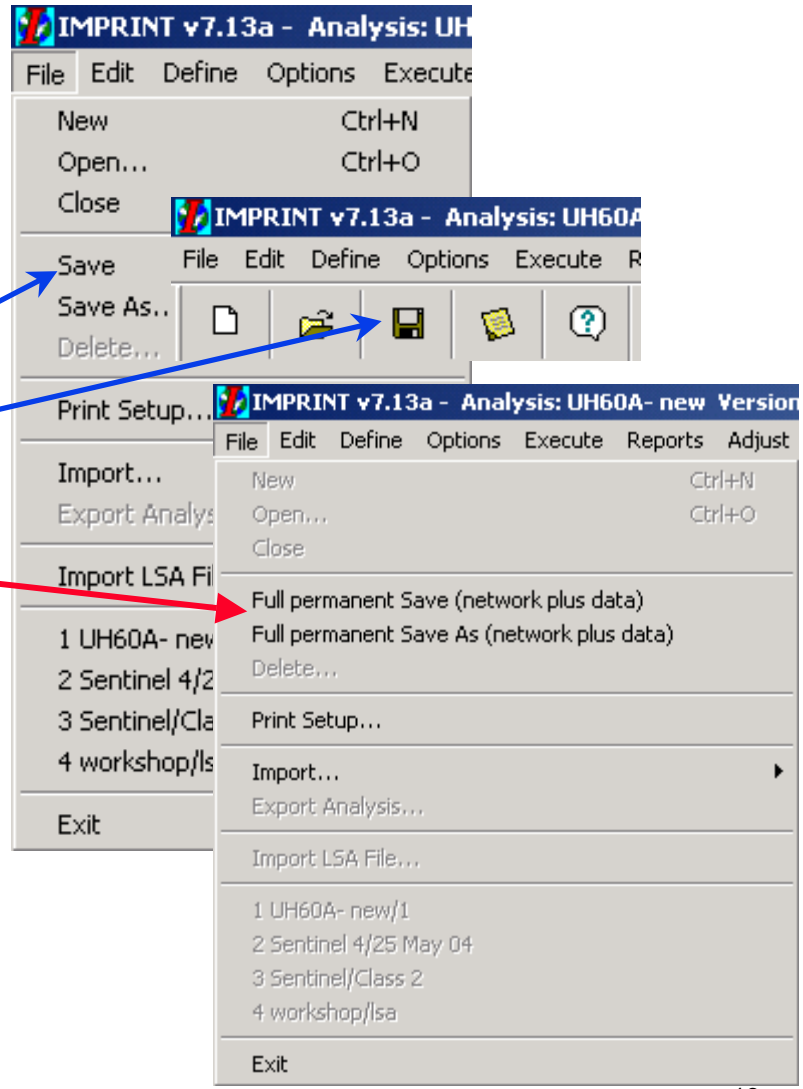
- What's in it
  - Executable files, & DLL files
  - IMPRINT database files
    - » "library" files - stuff that "comes with" IMPRINT
    - » "user" files - your stuff
    - » "working" or "session" files - for the open analysis
  - Report files - linked to an analysis
  - Help files
  - Documentation & Readme
    - » Analysis Guide & User's Guide



- What isn't: Your analysis by name!

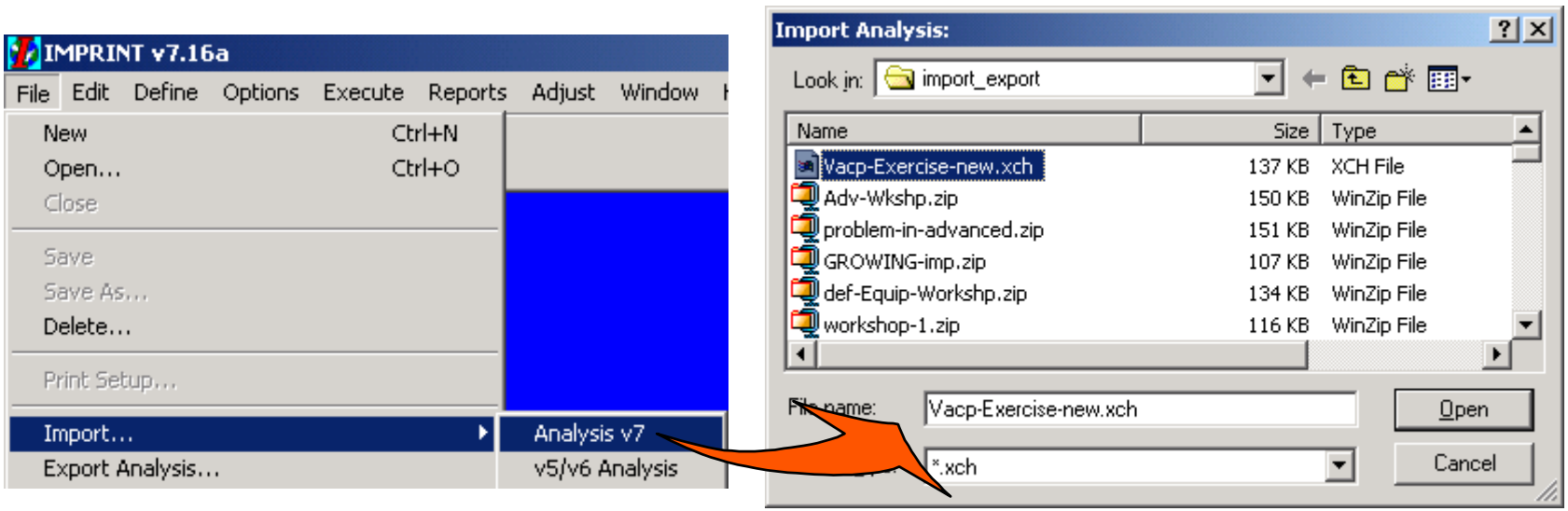
Saving Your Analysis

- Save early, save often*
  *from the top-most window
- Save again as you exit
- Saving your analysis
- Saving your network diagram & information
- When in doubt, save
- Reminders are legitimate!



Sharing Your Analysis Using Import & Export

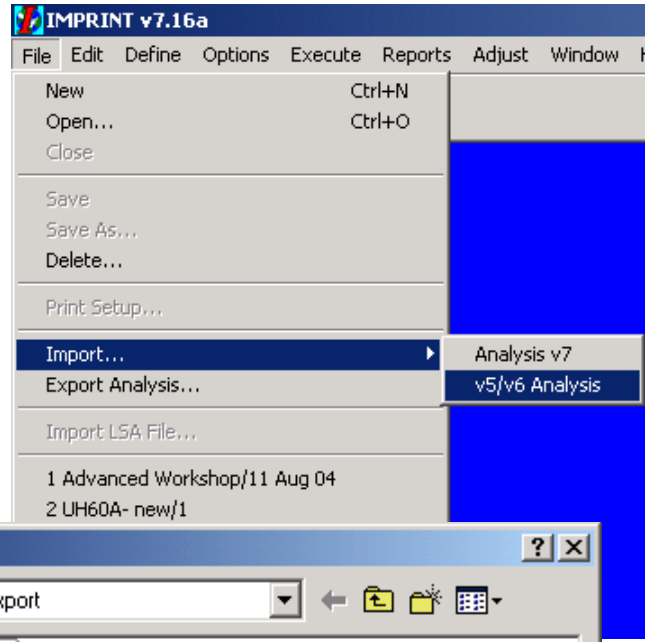
- To Import a version 7 analysis -
  - Close the open analysis
  - Select “Import - Analysis”
  - Browse until you find the one you’re looking for
  - To access the analysis, you must then open it

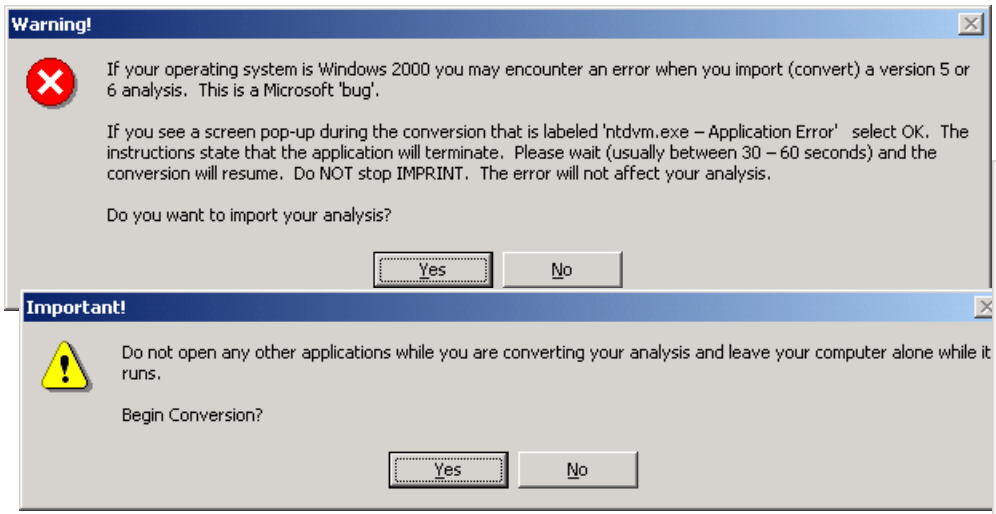


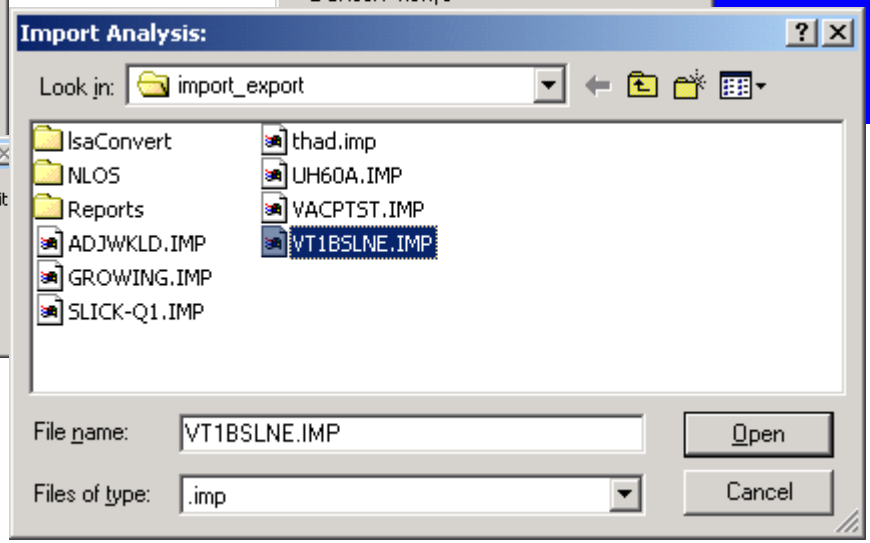
In IMPRINT, it’s an analysis. Out of IMPRINT, it’s a “.xch” file.

Sharing Your Analysis Using Import & Export (cont'd)

- To Import a version 5 or 6 analysis -
  - Close the open analysis
  - Select "Import – 5.0/6.0 Analysis"
  - Browse until you find the one you're looking for
  - To access the analysis, you must then open it

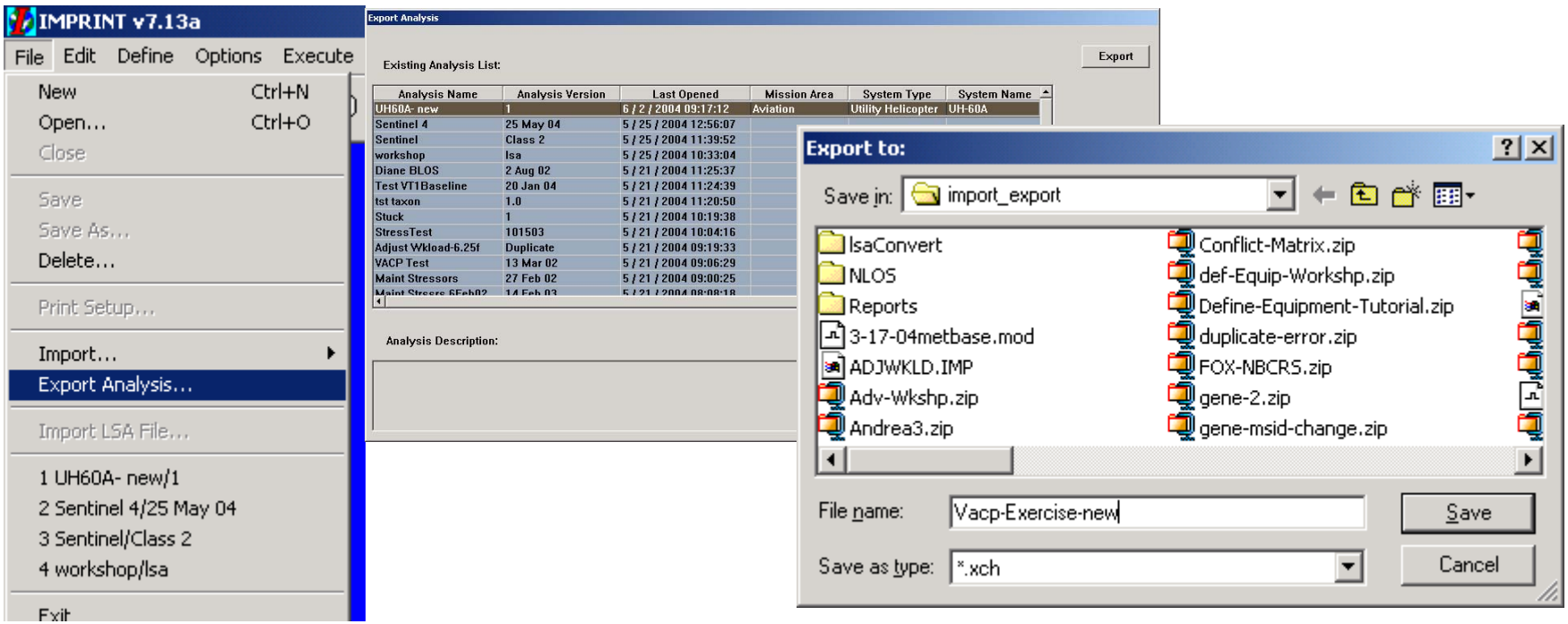






Sharing Your Analysis Using Import & Export (cont'd)

- To Export -
  - Close your analysis if you have one open
  - Select Export option
  - Create export file using Windows naming conventions
  - On hard drive or on disk
  - File name does not have to = analysis name

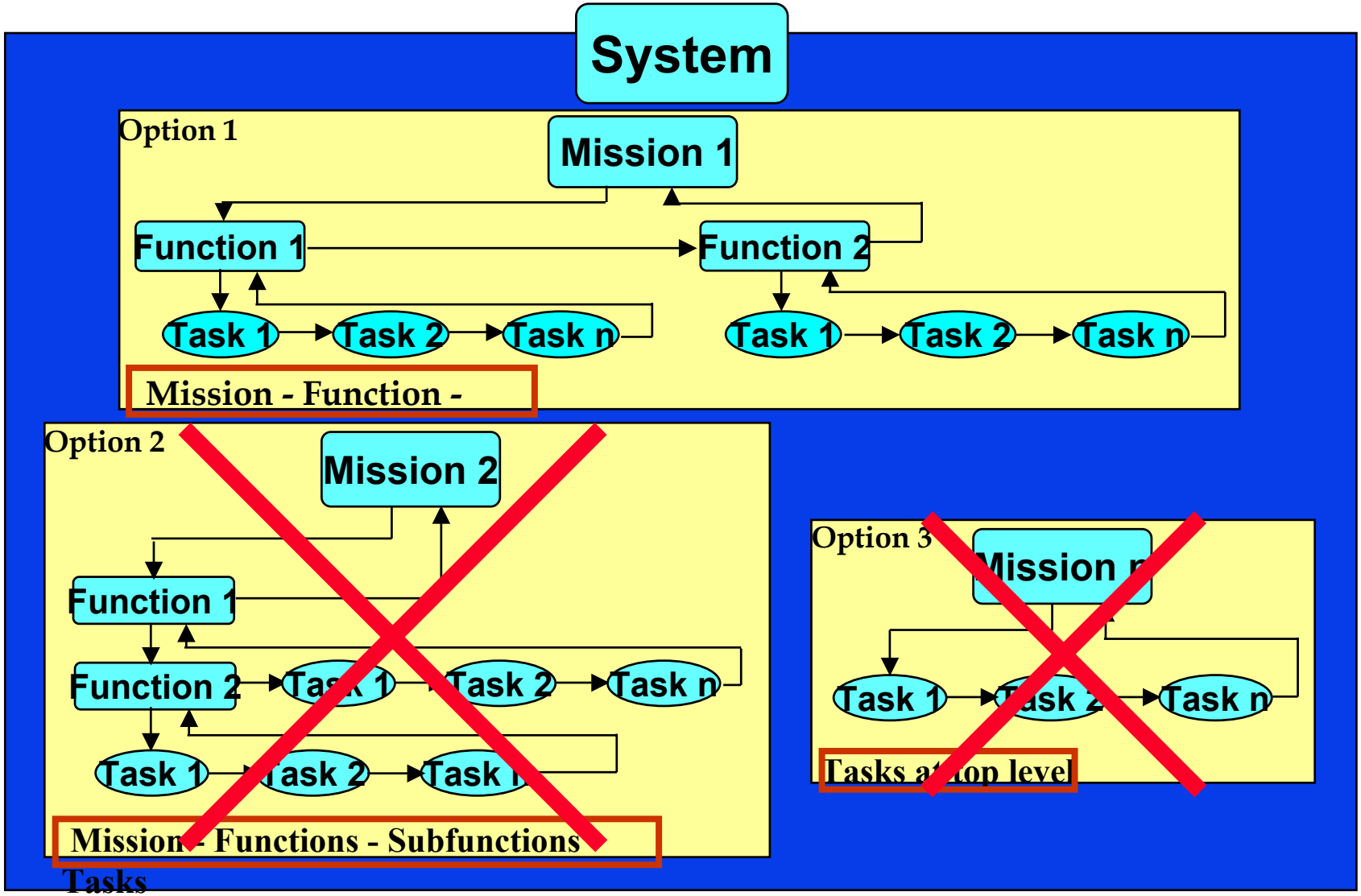


Introduction to Advanced Modeling

Advanced Modeling vs. Advanced workload

- Advanced modeling capabilities allow you greater flexibility in controlling the sequence of events in your model
  - Effects tab
- Advanced workload is another model for predicting workload based on multiple resource theory

Task Network Hierarchy Options in Advanced

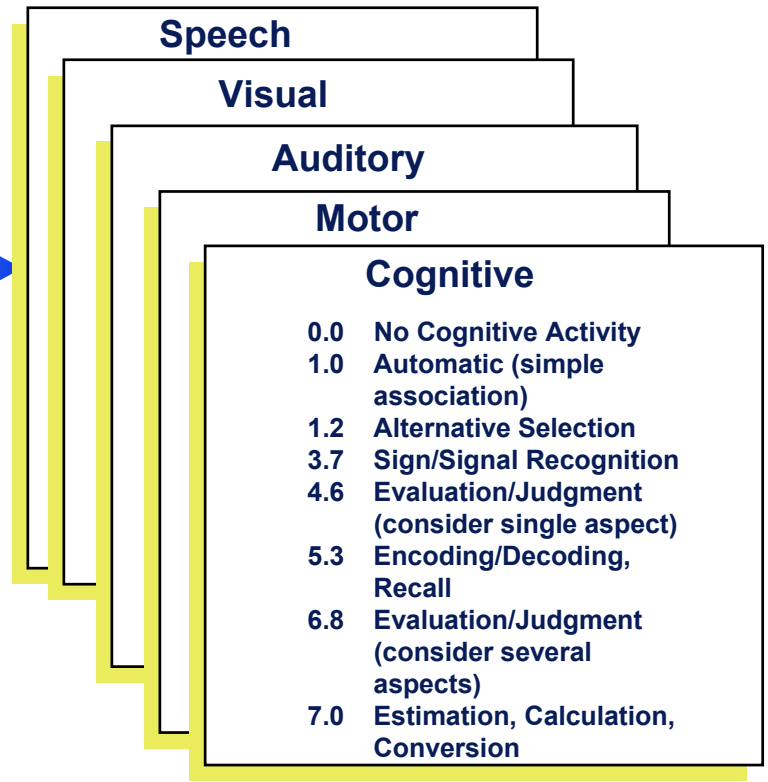


Multiple Resources Theory of Mental Workload

Mission Tasks → **Which Brain Resources Involved?** → **Degree of Resource Use?**

1. monitor alarms
2. decide response action
3. pull trigger
.
.
.
n. task n

- Visual
- Cognitive
- Auditory
- Motor
- Speech



Unique Outputs of Advanced Workload

- Critical Path
- Operator Activity
- Operator Workload
- Overload
- Channel Conflict
- Task Timeline
- CrewStation Workload
- User Snapshot

Branching Logic and Expressions

Advanced Modeling Details

- Variables
- Mathematical and Logical Expressions
- Beginning Effects
- Ending Effects
- Release Conditions
- Data Collection

Variables

- Most variables are defined by the user to represent conditions or parameters
  - Examples
    - » Resource availability
    - » Entity status
- Variable values can affect model execution
- Variable values are data to be collected

Variables

Similar to a cell in a spreadsheet or an algebraic expression

$$y = mx + b$$

	A	B	C	D
1				
2	Table			
3			Path1	
4				
5	Target			Speed
6				
7				

Variable Types

- Real
- Integer
- Arrays
  - Integer or real
  - 1, 2, or 3 dimensions
  - Indexed by a variable during a simulation
  - Think of them as tables with as many columns as you need

System Variables

- tag
  - A system variable that keeps track of the number of the entity being acted on by any event at any time
  - Once you assign a value, it gets carried through
  - In IMPRINT, tag denotes the operator number

System Variables

- clock
  - The current simulation time
  - Starts at zero
  - No expected resemblance to “real time”
- run
  - When you have multiple runs of the network, this tells you which run you’re in
  - Can be used to change conditions across runs

Naming Variables

- We recommend that variable names start in Caps
- System variables are usually lower case
- Variables are global and if you change a system variable unintentionally, it will impact the model run
- Exception is counters

SpeedRec[n] Target n clock

Expressions

- Expressions are used to initially set or change variable values in Micro Saint & WinCrew models
  - Examples
    - » To set and change resource availability
    - » To set and change entity states
- Two general types - mathematical and logical expressions

Mathematical Expressions

- Operators

( )	grouped operations
^	exponentiation
*	multiplication
/	division
%	remainder division
+	addition
-	subtraction

clock/60

Speed/LapDist

Laps*LapDist

n + 1

Advanced Expressions

- Logical operators

>	greater than
<	less than
>=	greater than or equal to
<=	less than or equal to
<>	not equal to
|	logical or
&	logical and
= =	equal to (e.g., if a== b then...)
: =	assignment

Advanced Expressions

- Logical statements

```
if
then
else
    if a>b then Time:=5 else Time:=7;
while ... do
    while i < 5 do Type[i] := 1, i=i+1;
```

Advanced Expressions

- Adjustment operators

+=	Increment by adding
- =	Increment by subtracting
/ =	Increment by dividing
* =	Increment by multiplying

n+=1 **m-=1**

n=n+1 **m=m-1**

Advanced Expressions

- Separators

; end of expression

e.g., a:=1;

, separates assignments within expressions

e.g., if a<1 then b:=5 , c :=6 else b:=4 , c:=2;

Comments

Comments are placed in curly brackets.

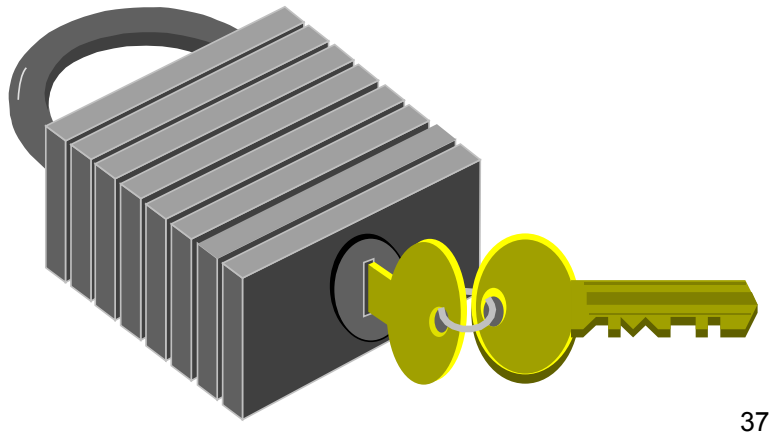
Note: All code must end in a ;

{This is an example of comments}1;

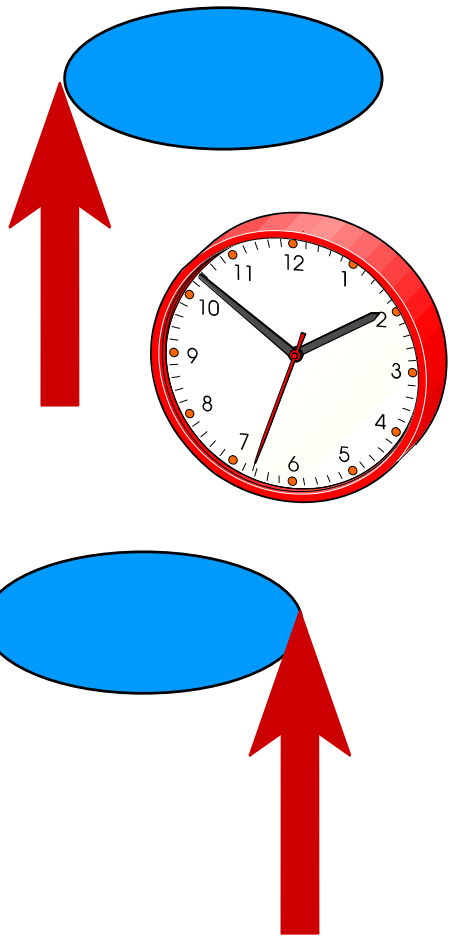
{When this task ends increment the task counter by 1} Taskcount+=1;

Release Conditions

- Allows task to execute
- Establishes rules for task execution
- Condition must be true - 1=true, 0=false
- Can include multiple conditions
- Typically involves logical expressions



Task Effects



- Beginning effects
  - What model conditions change as a result of a task beginning to execute
    - e.g., resources are used, counters are incremented
- Ending effects
  - What model conditions change as a result of a task completing execution
    - e.g., resources are free, counters are decremented

Order of Events

- MicroSaint executes events in this order
  - Release conditions
  - Mean time
  - Beginning Effects
  - Ending Effects
    - Succeeding task(s)

External Events

- Expressions that are inserted into model execution at selected clock times
- Can be used to cause task execution
- Used commonly to
  - Establish initial conditions of the system
  - Insert system changes (e.g., arrival rates)

“Start” and “Stop” Logic

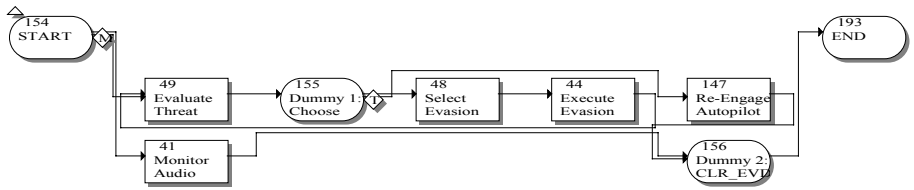
Develop Your Own Analysis

Develop Your Own Analysis

- Pick a Topic
- Develop a Question and Hypothesis

- Determine Measures

- Identify Functions and Tasks
- Build your Model



Does Your Model Run?

Yeah!!

So What???

Your Model Runs – So What?

- **Did it do what you wanted it to?**
  - First step is verification and debugging
- **How are you going to evaluate the results?**
  - Complete the analysis step
- **Is this realistic?**
  - Validate the model

VV&A or V(A)V&A

Verification, Validation, and Accreditation
vs.
Verification, Analysis, Validation, and Accreditation

- **Verification** means determining that the model does what it was meant to do
- **Analysis** of results means evaluating the results
- **Validation** means the model adequately represents the system
- **Accreditation** means that the model has be accredited for the use case

Debugging Tools

Debugging Hints

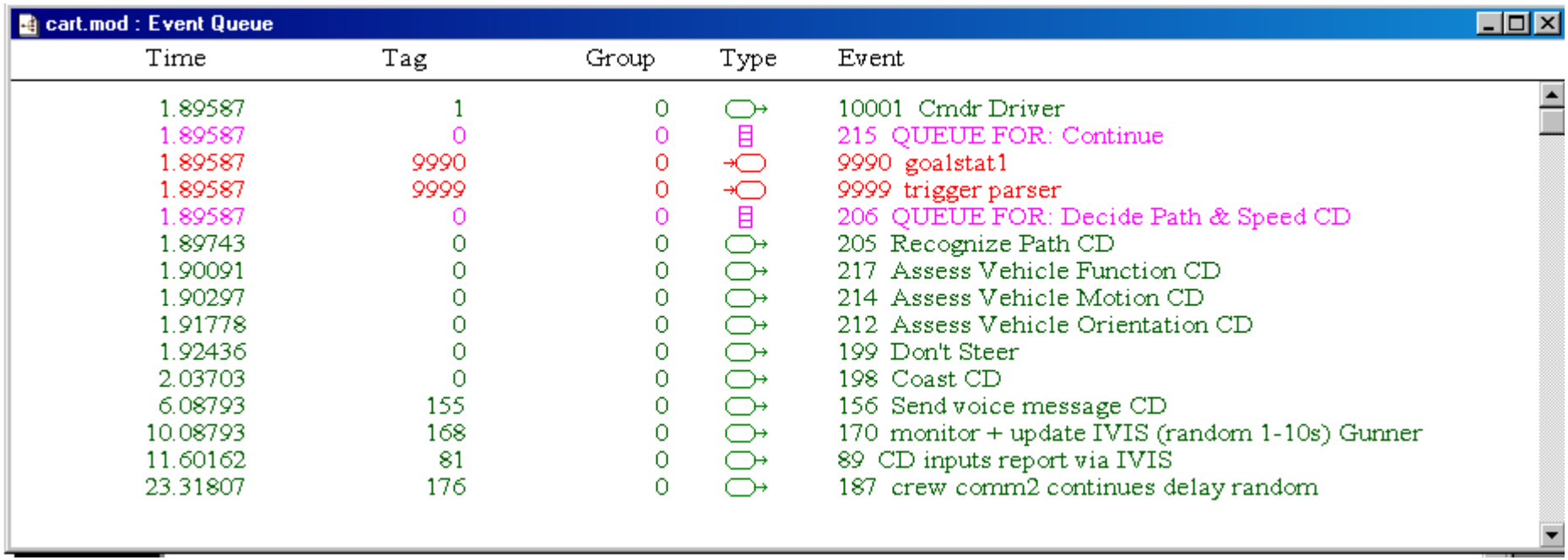
Step 1. Run model in IMPRINT
Step 2. Open the IMPRINT directory
Step 3. Open the correct version of MicroSaint
- VACP model – vacpsaint.exe
- Advanced model – crewsaint.exe
- Goal model – goalsaint.exe

Step 4. Open the model
- VACP model – imprint.mod
- Advanced model – saint.mod
- Goal model – cart.mod

Step 5. Run the model – Single step using Ctrl t

Note: No changes will be saved. You must go back and make the changes in IMPRINT.

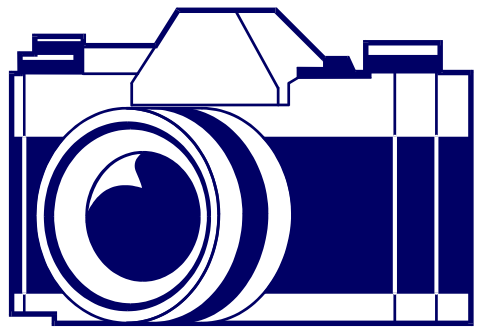
Event Queue


cart.mod : Event Queue

Time	Tag	Group	Type	Event
1.89587	1	0		10001 Cmdr Driver
1.89587	0	0		215 QUEUE FOR: Continue
1.89587	9990	0		9990 goalstat1
1.89587	9999	0		9999 trigger parser
1.89587	0	0		206 QUEUE FOR: Decide Path & Speed CD
1.89743	0	0		205 Recognize Path CD
1.90091	0	0		217 Assess Vehicle Function CD
1.90297	0	0		214 Assess Vehicle Motion CD
1.91778	0	0		212 Assess Vehicle Orientation CD
1.92436	0	0		199 Don't Steer
2.03703	0	0		198 Coast CD
6.08793	155	0		156 Send voice message CD
10.08793	168	0		170 monitor + update IVIS (random 1-10s) Gunner
11.60162	81	0		89 CD inputs report via IVIS
23.31807	176	0		187 crew comm2 continues delay random

Snapshots of Execution

- Used to collect data during a model run
- Each snapshot defines variables to be stored during execution
- Data collection can be triggered on
  - Task execution
  - Entities going into or departing from queues
  - Clock time
  - End of the run



What is mental workload?

Focus

What Is Mental Workload?

An Example

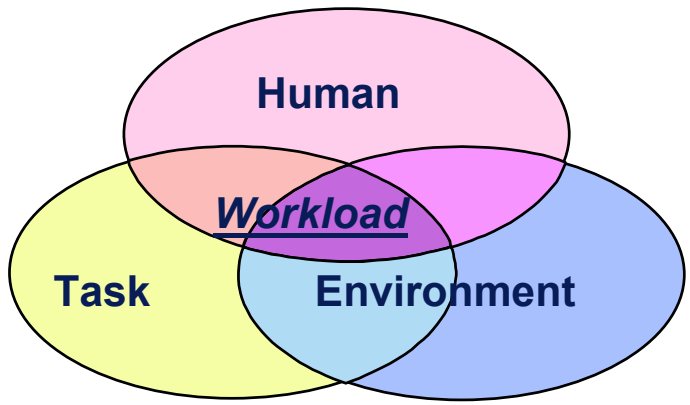
- Drivers slowing down to talk on their cell phone
- Accident rates of drivers using cell phones approaches that of drivers under the influence of alcohol

Why You Should Care About Workload

- If you reduce crewsize then some tasks must be automated or redistributed among remaining crew positions
  - Reallocation of tasks is likely to increase workload, thus increasing the potential for performance failures and errors.
  - Poorly designed automation can also increase workload and thus the potential for human errors.

Mental Workload

- Workload Definition
  - There is no universally agreed-upon definition
  - Today, however, there is generally agreement that, essentially, workload is
    - *the perceived relationship between the amount of mental processing capacity or resources and the amount required by the task*



*Gopher & Donchin, 1986, Hart & Staveland, 1988

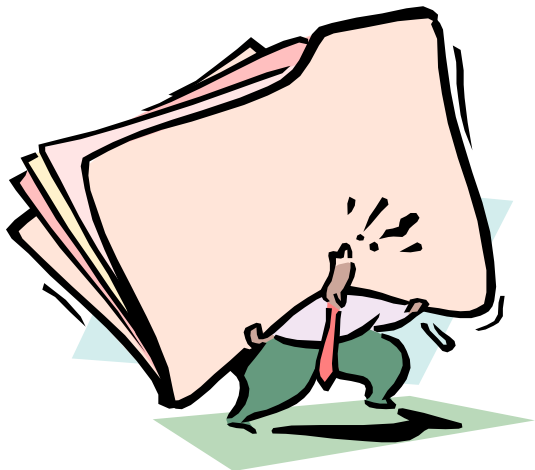
Mental Workload Issues

- Sustained low workload (underload) leads to boredom, loss of situation awareness, and reduced alertness.
- Sustained high workload (overload) leads to fatigue.
- Workload peaks lead to dropped tasks, increased task time, cognitive tunneling, and increased errors.
- These factors reduce crew performance, system performance, and contribute to mission failure

Mental Workload Objective

Achieve evenly distributed, manageable workload.

Avoid both overload and underload.



Various Mental Workload Measurement Approaches

empirical

- physiological
- primary task
- secondary task
- subjective rankings

analytical

- workload modeling-IMPRINT

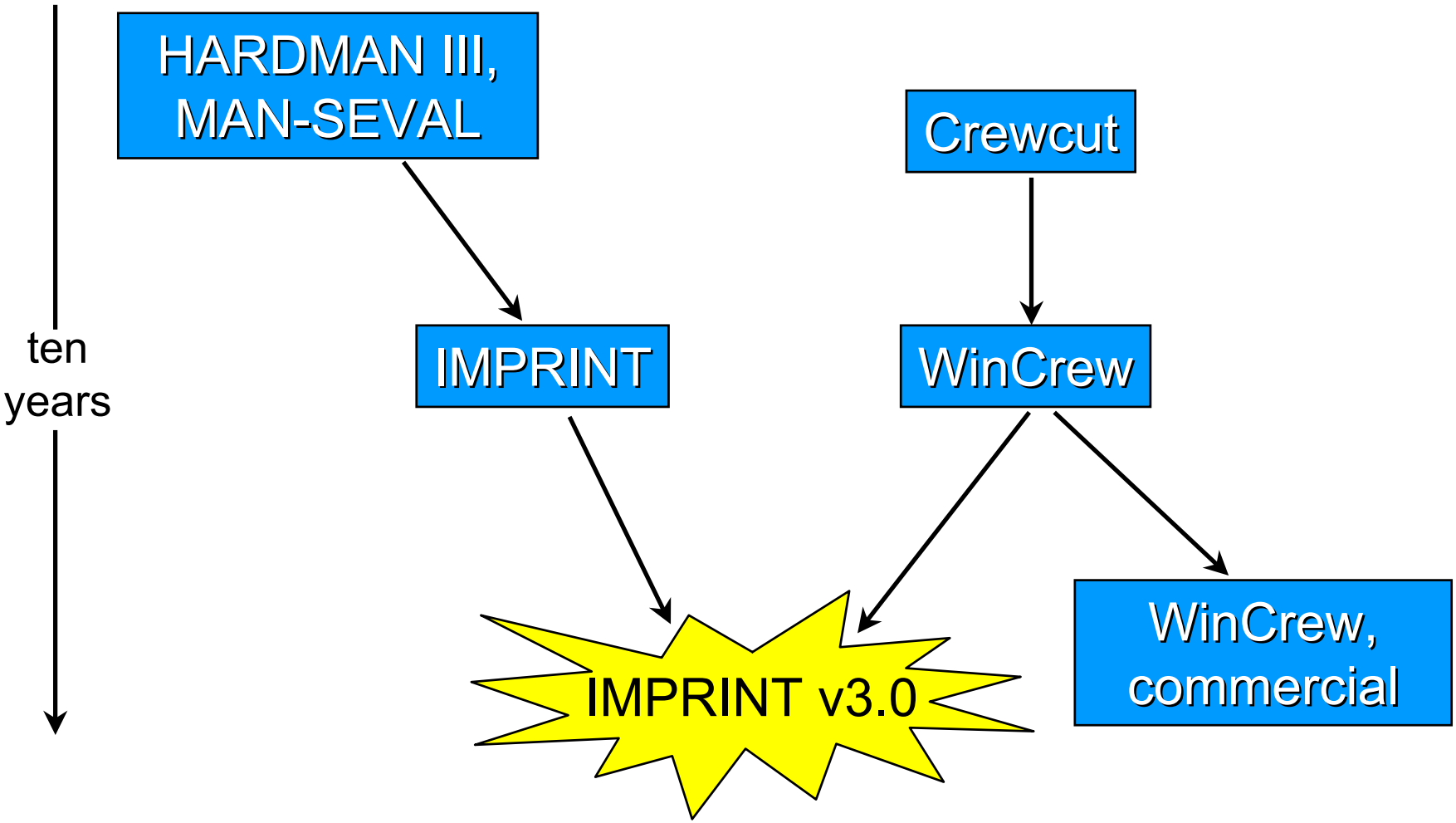
Workload Modeling

- Workload modeling of human behavior is a technique that has been used to *predict* workload levels. Many different models have been proposed to model human behavior and measure workload.

Workload Modeling

- Workload modeling of human behavior is a technique that has been used to *predict* workload levels.
  - IMPRINT can be used to model and predict mental workload.

ARL HRED Workload Modeling Tools



IMPRINT Workload Modeling

- workload predictions are used to estimate operator performance and its effect on system performance

What is advanced IMPRINT workload?

Multiple Resources Theory of Mental Workload

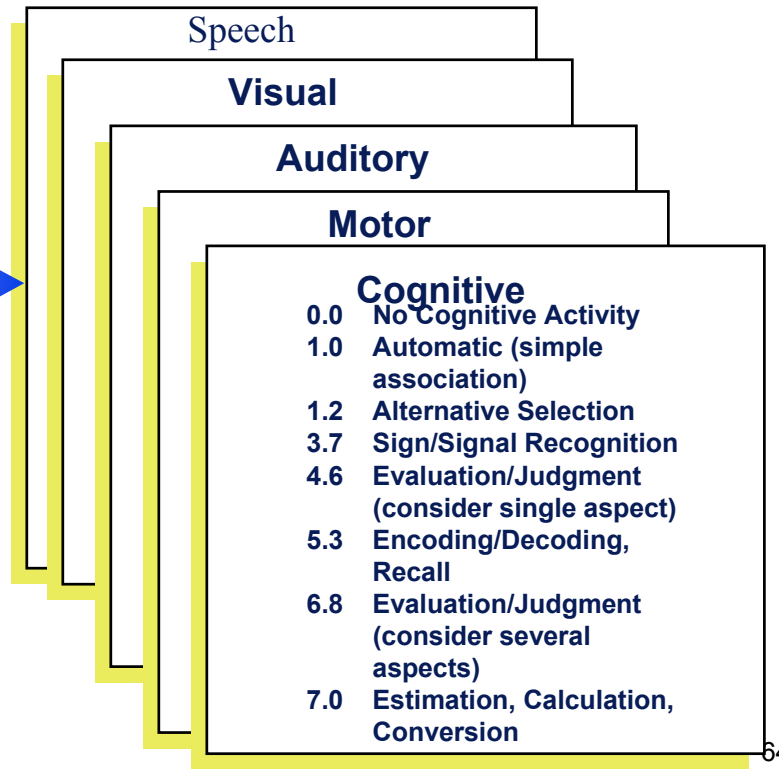
Mission Tasks → **Which Brain Resources Involved?** → **Degree of Resource Use?**

1. monitor alarms
2. decide response action
3. pull trigger

.
.
.

n. task n

- Visual
- Cognitive
- Auditory
- Motor
- Speech



Aggregate Workload

ADVANCED WORKLOAD CALCULATION:

$$W_T = W_{STD} + (W_{WCC} + W_{BCC})$$

Where:

W_T = Instantaneous Workload at Time T

W_{STD} = Workload attributable to the demands of all operator's tasks at time T (Single Task Demands)

W_{WCC} = Workload attributable to Within-Channel Conflicts (Within and between tasks)

W_{BCC} = Workload attributable to Between-Channel Conflicts (Between tasks only; within tasks may see improved performance "S-C-R")

* Adapted from W/Index North & Riley, 1988

Workload & Crewstation Parameters

Define Resources and Interfaces

Resources

- Default set
  - Visual
  - Auditory
  - Motor
  - Speech
  - Cognitive
- Add up to 5 more

Interfaces

- Controls and displays in your design
- Don't get carried away!

Be prudent!

Assign Single Task Demands

- Pair resources with interfaces
- Assign resource interface pairs to tasks
- Estimate single task demands for tasks

  Pop-up scales accessed by double-clicking the cell

Assign Conflicts

- BIG contributor to workload score
- Penalties for using resource interface pairs
- Cell values range from 0.00 -1.00
- Double click on row header provides "expert guidance" and default values

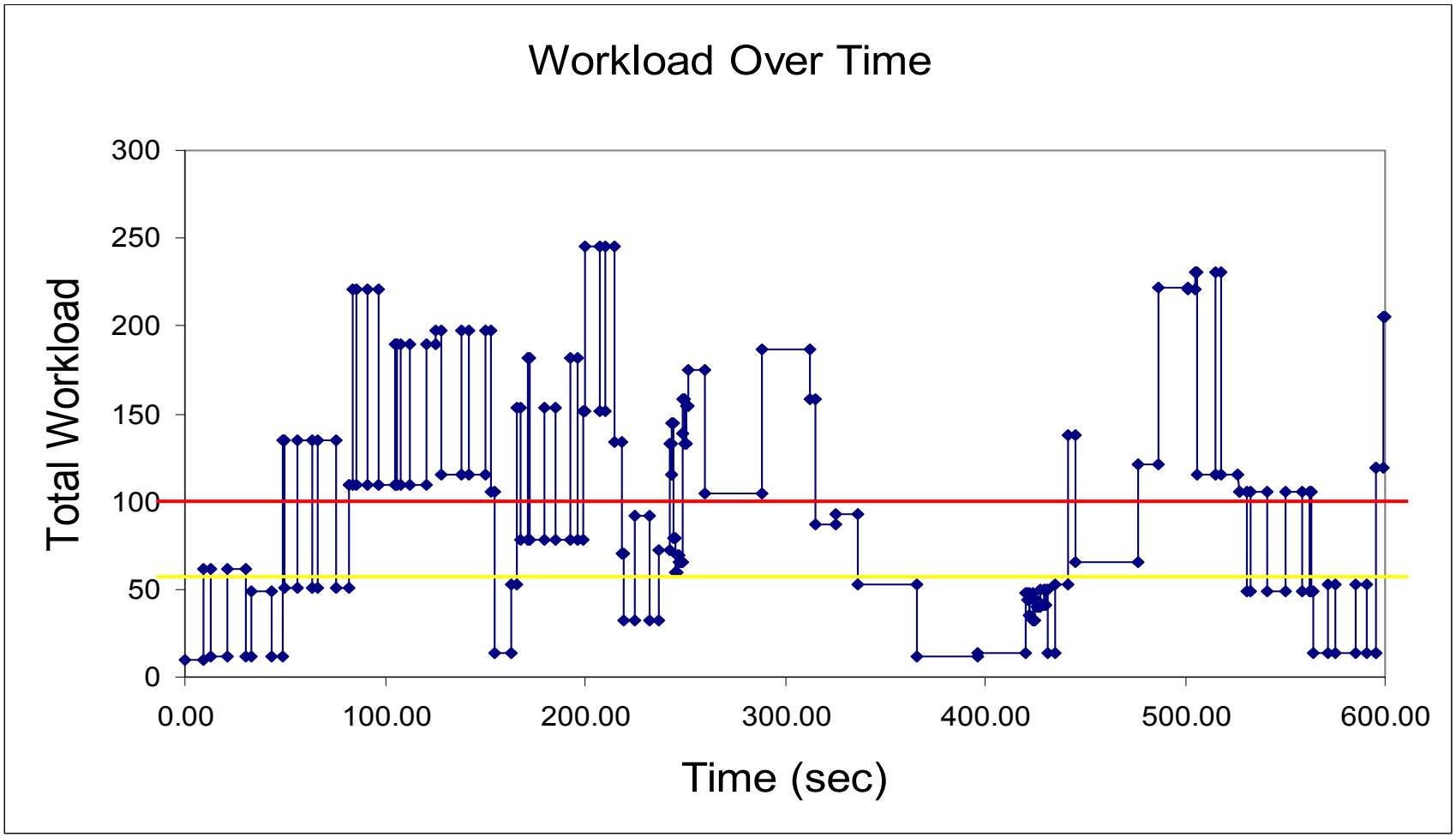
Advanced Workload Method

- Describes effort needed to perform task
- To help examine impact of workload during mission
- Results are combined across channels into one score
- Results consider inter- & intra-channel conflict
- Does dynamically impact performance

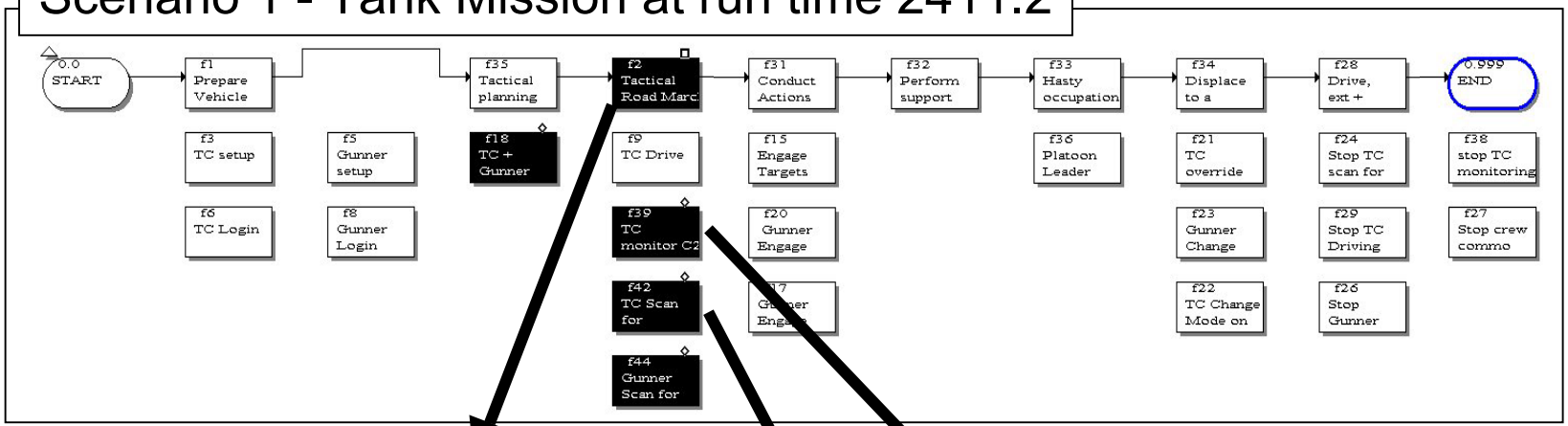
Time, Accuracy, Crew Allocation, Sequence

Workload Threshold

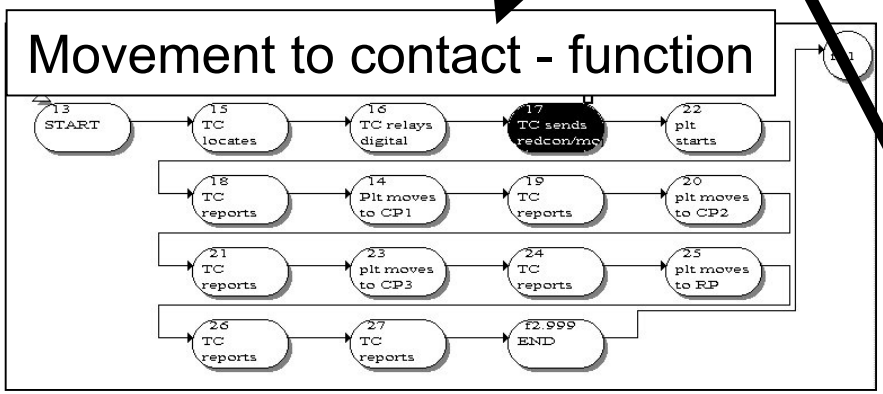
FCS Modeling Results CG



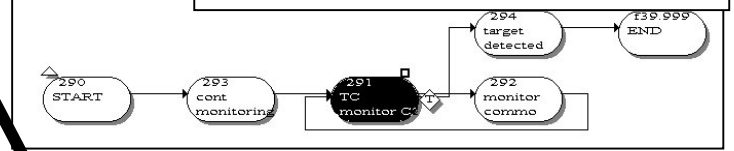
Scenario 1 - Tank Mission at run time 2411.2



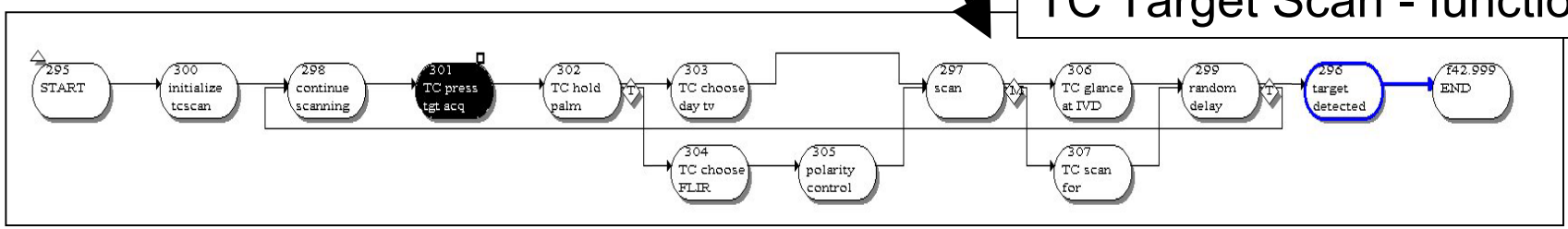
Movement to contact - function



Monitor C2 - function



TC Target Scan - function

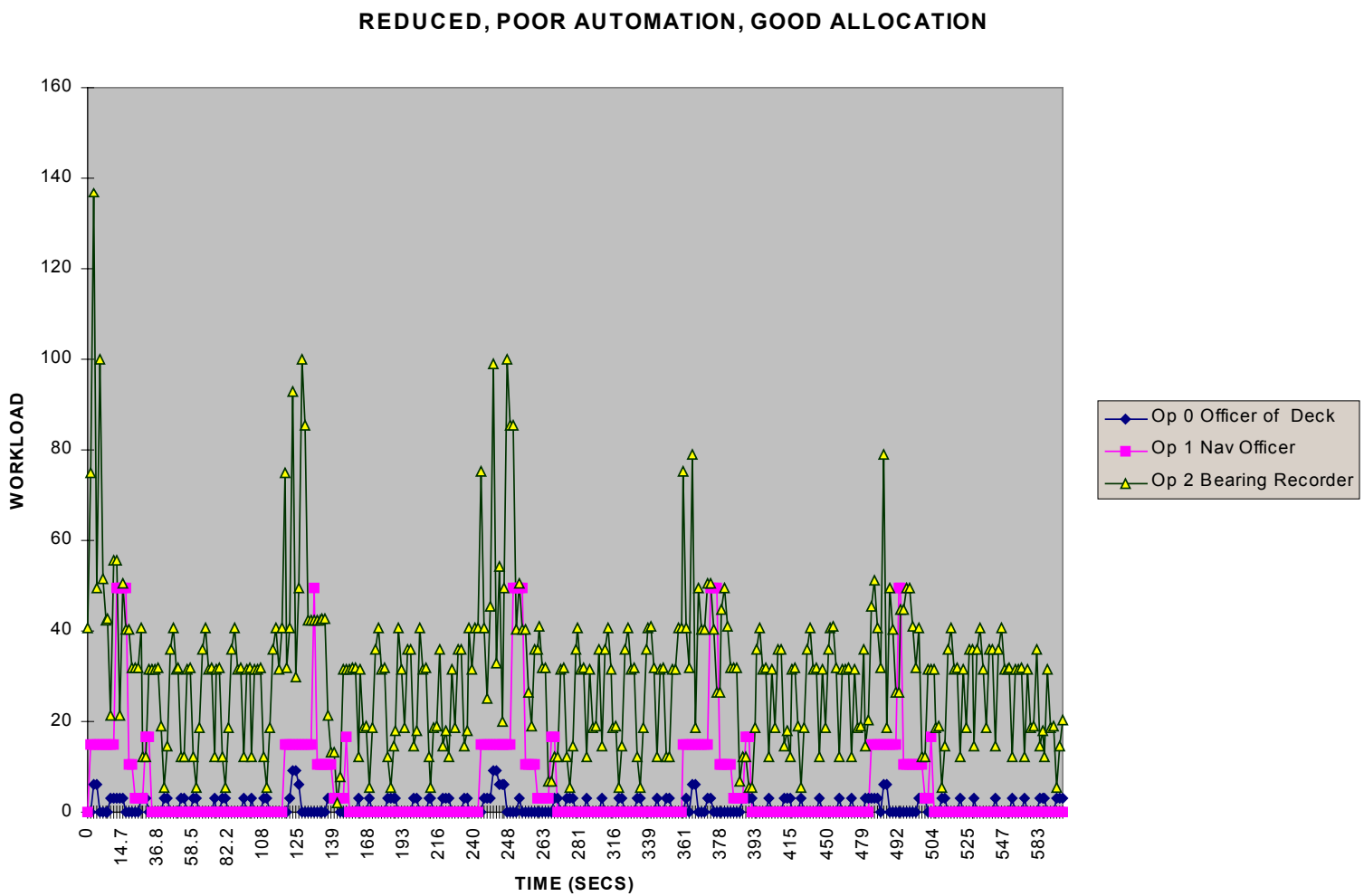


Analysis of Results

Unique Outputs of Advanced Workload

- Critical Path
- Operator Activity
- Operator Workload
- Overload
- Channel Conflict
- Task Timeline
- CrewStation Workload
- User Snapshot

Sample WinCrew Output



Displaying your results

- Consider your audience
- Display your results accordingly
  - Spreadsheets
  - Tables
  - Graphs

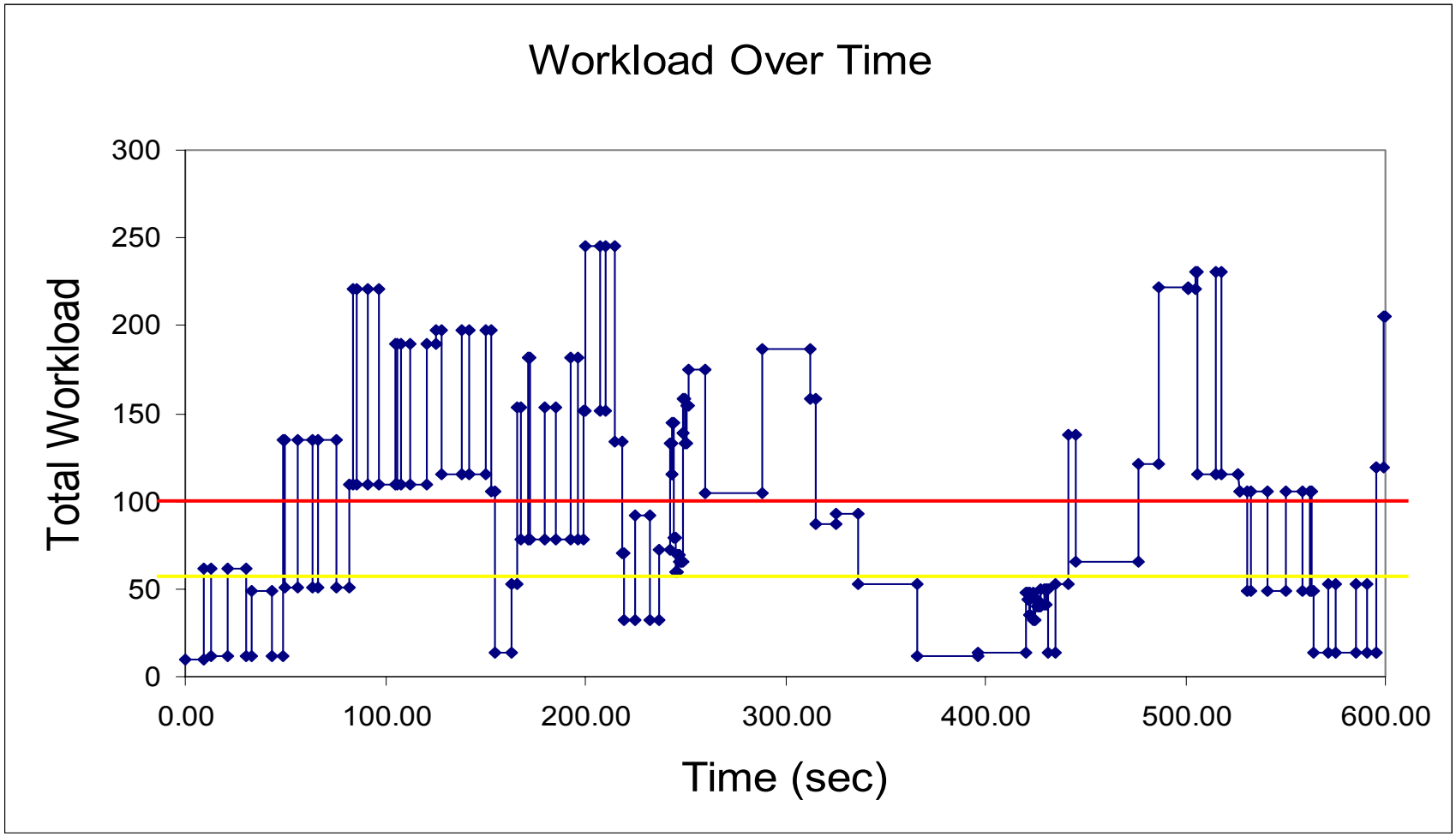
Results - spreadsheets

Thresholds		Results							
T1 =	60	Number of Times over	60	97		Number of Times over	100	59	
T2 =	100	Percent of time in	60	39.8%		Percent of time in overload	100	24.0%	
		Max Workload Value		283.9		Min Workload Value		7	
		T1				T2			
Time	Workload	>?	Secs >			>?	Secs>		
0	14.7	0	0			0	0		
3	58.01	0	0			0	0		
8	13.5	0	0			0	0		
9	54.6	0	0			0	0		
12	32.11	0	0			0	0		
14	38.67	0	0			0	0		
15.89	42.09	0	0			0	0		
16	106.91	1	2			1	2		
18	244.18	1	1			1	1		
19	155.93	1	2			1	2		
21	7	0	0			0	0		
23	29.66	0	0			0	0		
25	48.96	0	0			0	0		
25.12	99.93	1	3.88			0	0		
29	13.2	0	0			0	0		
31	14.7	0	0			0	0		
31.06	57.86	0	0			0	0		
34	76.89	1	0.69			0	0		
34.69	163.74	1	2.31			1	2.31		
37	69.99	1	2			0	0		

Results - tables

	Combat	AutoScan	Baseline	Non-Combat
	T1 = 60	T1 = 60	T1 = 60	T1 = 60
Number of Times over T1	112	91	97	50
Percent of time in overload > T1	46.9%	37.7%	39.8%	34.2%
Max Workload Value	244.18	287.02	283.9	245.82
Min Workload Value	7	3	7	7
	T2 = 100	T2 = 100	T2 = 100	T2 = 100
Number of Times over T2	69	44	59	29
Percent of time in overload > T2	28.3%	19.1%	24.0%	23.5%

Results - graphs

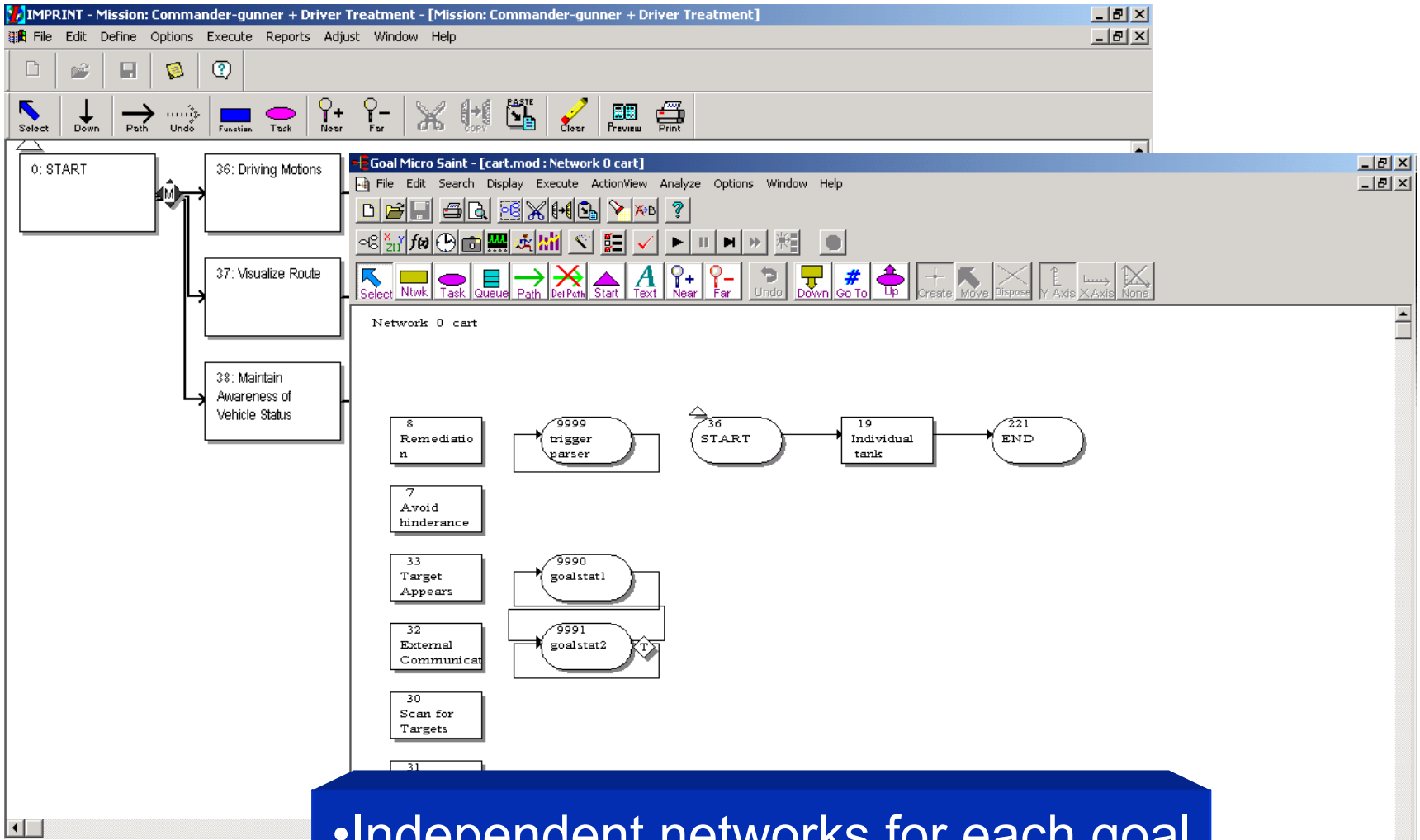


Goal Oriented Modeling

Goal Oriented Modeling

- Goal orientation
  - Option from VACP
  - Beginning & Ending Effects
  - Variable Catalog
  - Macros (User-Defined Functions)
  - Snapshots
- COM capabilities
  - Including HLA Middleware
- Access to tag

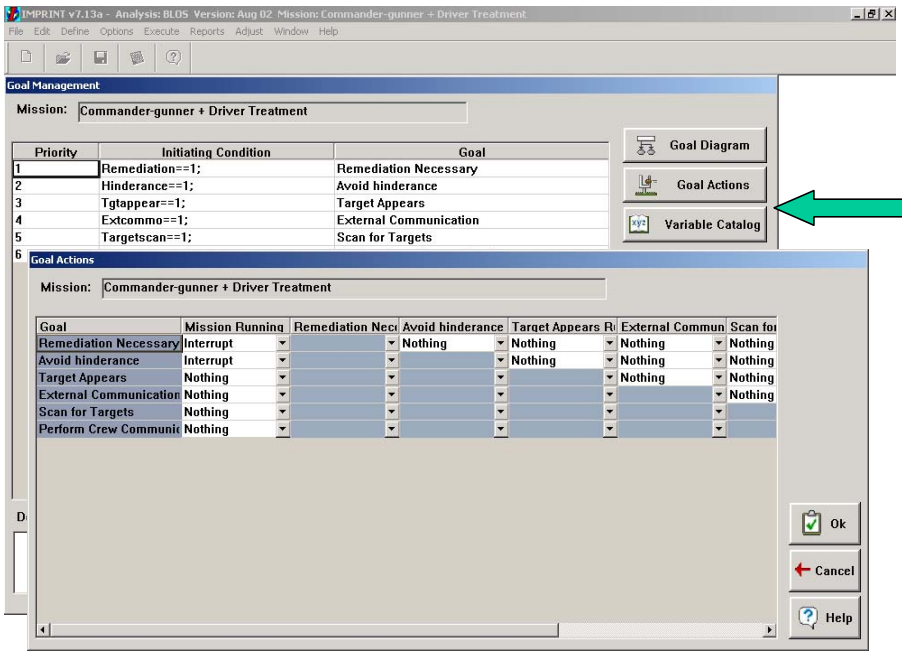
Task Network Model Development

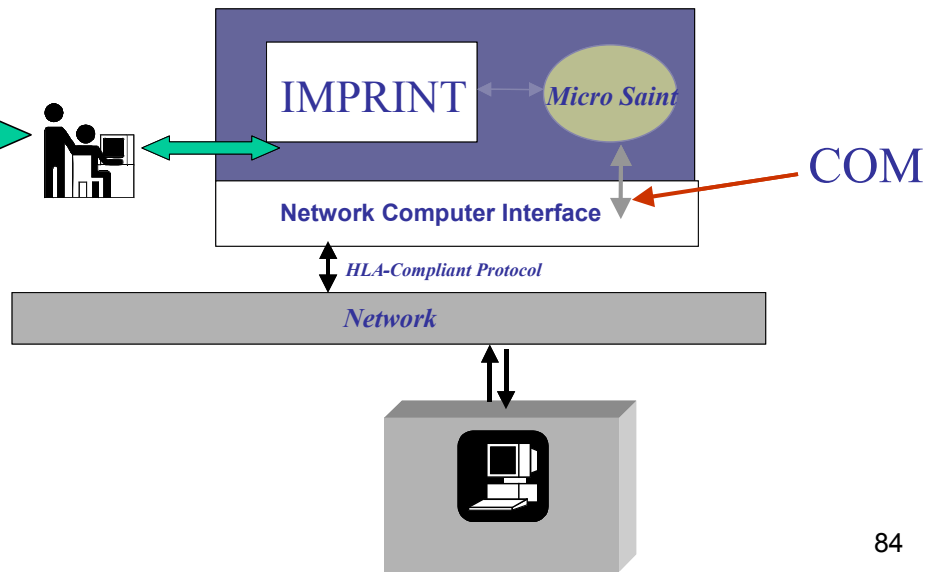


- Independent networks for each goal
- Variables are still global

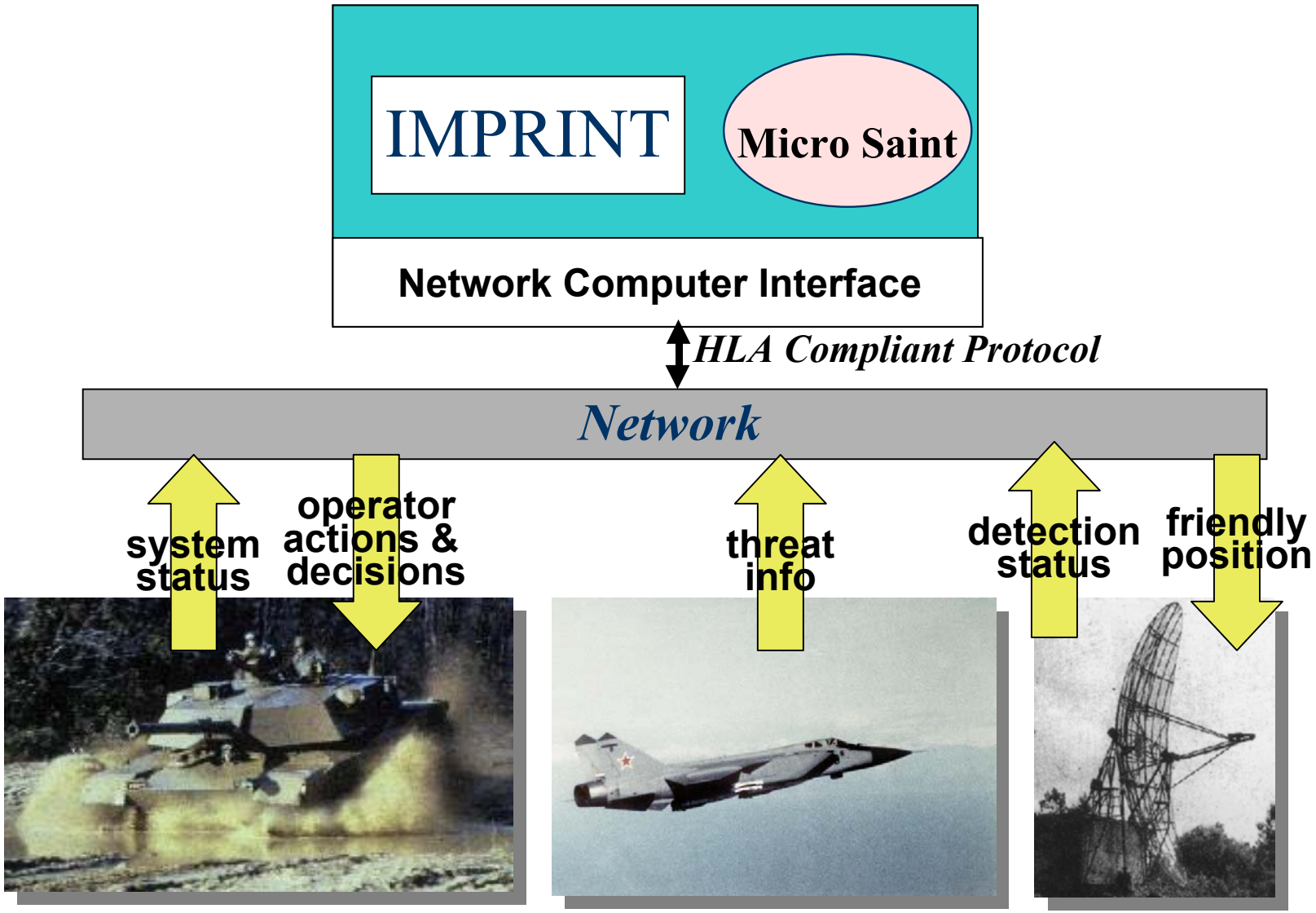
Goal Functions

- Trigger identification
- Trigger communication
- Task interruption
- Task restart vs. task resume





System Architecture



News of Note

- AF Validation Success Story
  - Wright Pat SIMAF Virtual Strike Warfare Environment
  - Time critical targeting (SCUD Hunt) mission
  - HPM vs. Eight pilots (F16 and A10)
  - Overall kills of ground targets in the time critical scenario was virtually the same for both the model and pilots (100% and 98%, respectively)
  - HPM accounted for 61 percent of the behavior of the pilots in the simulation environment
  - New tactic discovered: Coordinated use of synthetic aperture radar (SAR) and targeting infrared (TIR) imaging system

Why would you use Goal-Oriented?

1. **When you want VACP workload and the ability to use effect modeling**

2. **When you want to represent human behavior using goals**

3. **When you need to talk to other simulations**

You can switch from VACP or Advanced to Goal oriented with caveats!

Review Advanced Modeling Capabilities

Advanced Modeling Details

- Variables
- Mathematical and Logical Expressions
- Beginning Effects
- Ending Effects
- Release Conditions
- Data Collection

Comments

Comments are placed in curly brackets.

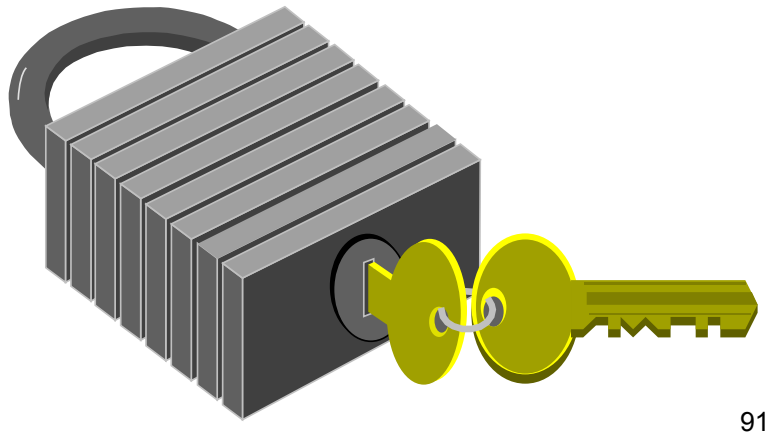
Note: All code must end in a ;

{This is an example of comments}1;

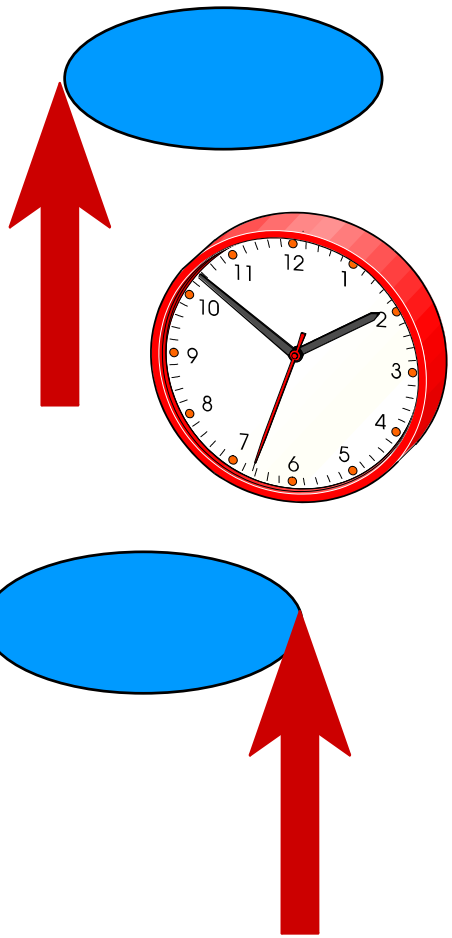
{When this task ends increment the task counter by 1} Taskcount+=1;

Release Conditions

- Allows task to execute
- Establishes rules for task execution
- Condition must be true - 1=true, 0=false
- Can include multiple conditions
- Typically involves logical expressions



Task Effects



- Beginning effects
  - What model conditions change as a result of a task beginning to execute
    - e.g., resources are used, counters are incremented
- Ending effects
  - What model conditions change as a result of a task completing execution
    - e.g., resources are free, counters are decremented

Order of Events

- MicroSaint executes events in this order
  - Release conditions
  - Mean time
  - Beginning Effects
  - Ending Effects
    - Succeeding task(s)

External Events

- Expressions that are inserted into model execution at selected clock times
- Can be used to cause task execution
- Used commonly to
  - Establish initial conditions of the system
  - Insert system changes (e.g., arrival rates)

Debugging Hints

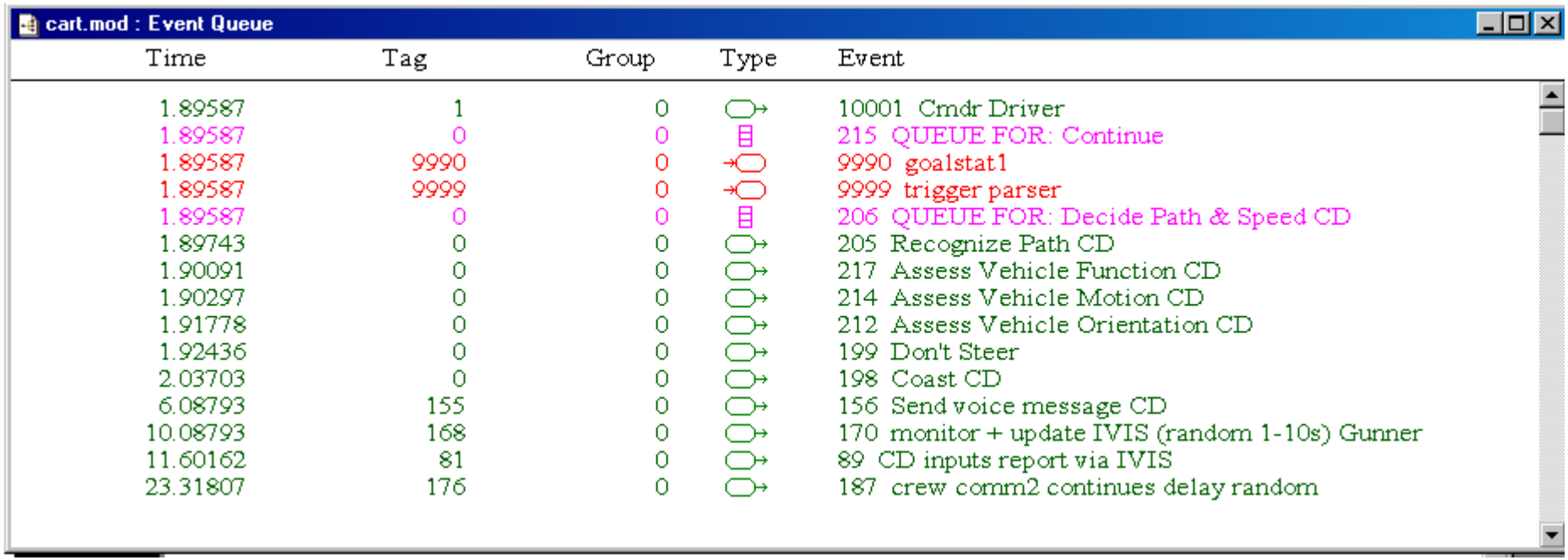
Step 1. Run model in IMPRINT
Step 2. Open the IMPRINT directory
Step 3. Open the correct version of MicroSaint
- VACP model – vacpsaint.exe
- Advanced model – crewsaint.exe
- Goal model – goalsaint.exe

Step 4. Open the model
- VACP model – imprint.mod
- Advanced model – saint.mod
- Goal model – cart.mod

Step 5. Run the model – Single step using Ctrl t

Note: No changes will be saved. You must go back and make the changes in IMPRINT.

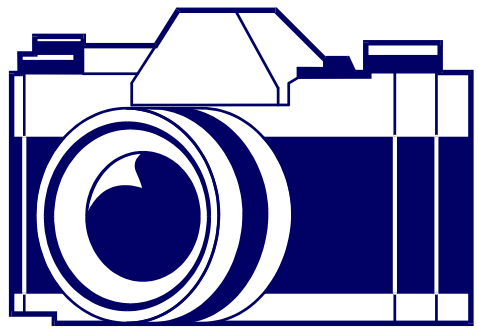
Event Queue



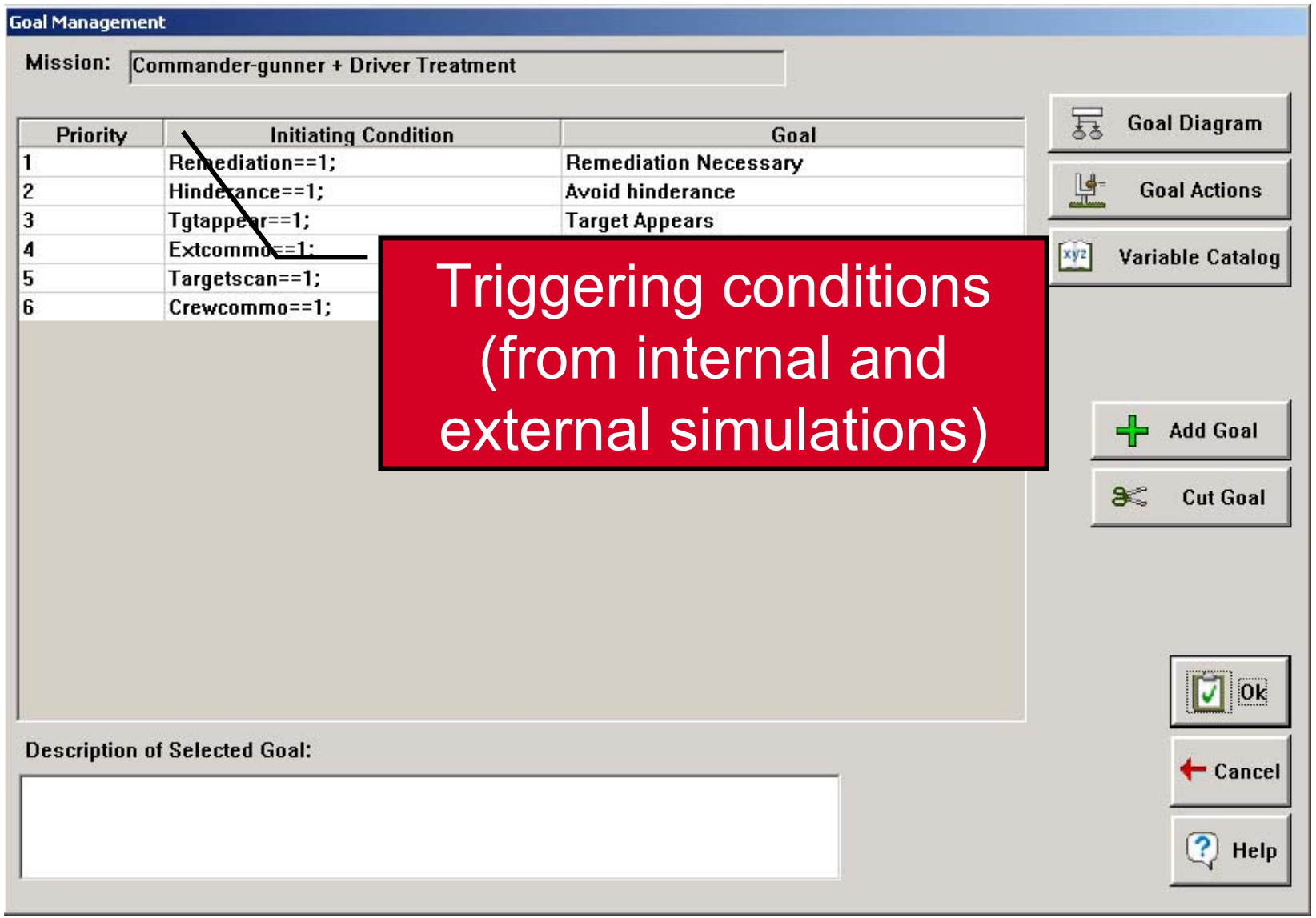
Time	Tag	Group	Type	Event
1.89587	1	0		10001 Cmdr Driver
1.89587	0	0		215 QUEUE FOR: Continue
1.89587	9990	0		9990 goalstat1
1.89587	9999	0		9999 trigger parser
1.89587	0	0		206 QUEUE FOR: Decide Path & Speed CD
1.89743	0	0		205 Recognize Path CD
1.90091	0	0		217 Assess Vehicle Function CD
1.90297	0	0		214 Assess Vehicle Motion CD
1.91778	0	0		212 Assess Vehicle Orientation CD
1.92436	0	0		199 Don't Steer
2.03703	0	0		198 Coast CD
6.08793	155	0		156 Send voice message CD
10.08793	168	0		170 monitor + update IVIS (random 1-10s) Gunner
11.60162	81	0		89 CD inputs report via IVIS
23.31807	176	0		187 crew comm2 continues delay random

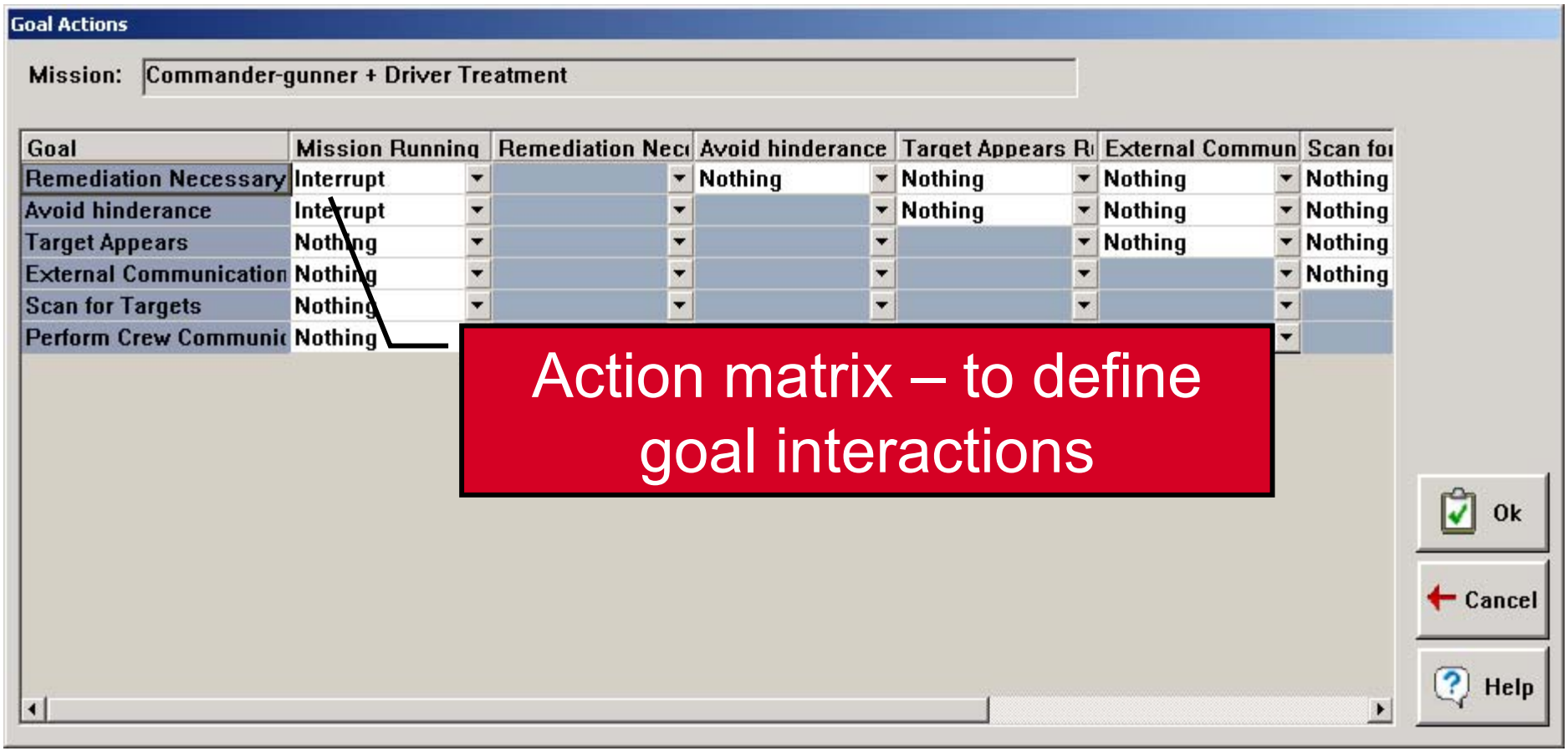
Snapshots of Execution

- Used to collect data during a model run
- Each snapshot defines variables to be stored during execution
- Data collection can be triggered on
  - Task execution
  - Entities going into or departing from queues
  - Clock time
  - End of the run



Goal Matrix


Goal Management
Mission:
Commander-gunner + Driver Treatment
Priority
Initiating Condition
Goal
1
Remediation==1;
Remediation Necessary
2
Hinderance==1;
Avoid hinderance
3
Tgtappear==1;
Target Appears
4
Extcommo==1;
5
Targetscan==1;
6
Crewcommo==1;
Goal Diagram
Goal Actions
Variable Catalog
Triggering conditions (from internal and external simulations)
Add Goal
Cut Goal
Ok
Cancel
Help
Description of Selected Goal:


Goal Actions
Mission: Commander-gunner + Driver Treatment
Goal
Mission Running
Remediation Nec
Avoid hinderance
Target Appears R
External Commun
Scan for
Remediation Necessary
Interrupt
Nothing
Nothing
Nothing
Nothing
Avoid hinderance
Interrupt
Nothing
Nothing
Nothing
Nothing
Target Appears
Nothing
Nothing
Nothing
Nothing
External Communication
Nothing
Nothing
Nothing
Scan for Targets
Nothing
Perform Crew Communic
Nothing
Action matrix – to define goal interactions
Ok
Cancel
Help

Goal Rules

- **When a trigger comes true:**
  - Look UP the matrix to see if a higher priority goal would suspend or halt it. If so, don't start it, but keep trying.

    If not:
  - Look DOWN the matrix and implement the actions for all lower priority goals

- **When a goal ends normally, gets halted or gets suspended:**
  - Resume anything it suspended UNLESS a higher priority goal would halt it. If so, halt it. If a higher priority goal would suspend it, then suspend it.

Review of IMPRINT Capabilities

Using IMPRINT

IMPRINT MODULES		DEFINE MISSION			DEFINE EQUIPMENT	DEFINE SOLDIER	DEFINE SUPPLY
		VACP Modeling	Advanced Modeling	Goal Oriented			
CAPABILITIES	**Interface Technique**						
	Data Entry				√	√	√
	Graphical Toolbar	√	√	√			
	Embedding Coding		√	√			
	Additional Capabilities						
	VACP Workload	√		√			
	Advanced Workload		√				
	PTS	√		√	√		

Wrap up and Discussion!

Lessons Learned

- Save ! Save! Save!
- Never too many DUMMIES…
- Naming Conventions

Getting the Software

Who

- Any government agency
- Private industry with government contract
- Foreign government (case-by-case)

How

- Send request via e-mail or letter
- If private industry include government contract number and organization

Non-Distribution Form

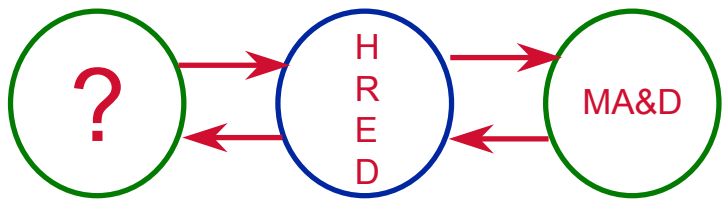
- Keep track of users
- Reminder not to distribute

Software Distribution

Technical Support

ARL-HRED

- Ms. Celine Richer (cricher@arl. army. mil)
  (410) 278-5883
- Ms. Diane Mitchell (diane@arl. army. mil)
  (410) 278-5878
- Ms. Jody Wojciechowski (jqw@arl.army.mil)
  (410) 278-8830
- Ms. Charneta Samms (csamms@arl.army.mil)
  (410) 278-5877



Maintain Database

- User comments
- “Bugs”
- “Fixes”

Using the List Server

List of current IMPRINT users & interested parties

Send suggestions, comments, general information or questions regarding IMPRINT to

imprint@arl.army.mil

References

- Allender, L., Kelley, T. D., Salvi, L., Lockett, J., Headley, D. B., Promisel, D., Mitchell, D., Richer, C., and Feng, T. Verification, validation, and accreditation of a soldier-system modeling tool. Proceedings of the Human Factors and Ergonomics Society 39th Annual Meeting-1995, San Diego, pp. 1219-1223.

- Allender, L., Salvi, L., and Promisel, D.(June 1997). Evaluation of Human Performance under Diverse Conditions via Modeling Technology. Proceedings of Workshop on Emerging Technologies in Human Engineering Testing and Evaluation, NATO Research Study Group 24. Brussels, Belgium.

- Allender, L., Kelley, T., Archer, S., and Adkins, R., (1997). IMPRINT The Transition and Further Development of a Soldier-System Analysis Tool. MANPRINT Quarterly,Office of the Deputy Chief of Staff of Personnel, Vol. V, No. 1.

- Dahl, S., Allender, L., and Kelley, T., (1995) Transitioning Software to the Window Environment - Challenges and Innovations. Proceedings of the Human Factors and Ergonomics Society 39th Annual Meeting - 1995, San Diego, pp. 1224-1227.

- McMahon, R., Spencer, M., and Thornton, A. (1995). A quick response approach to assessing the operation performance of the XM93E1 NBCRS through the use of modeling and validation testing. Presented at the Military Operations Research Society Symposium.

- Micro Analysis & Design. Stressor Review Report: Enhanced Performance Degradation Factors and Upgrades for Improved Performance Research Integration Tool (IMPRINT) Version 5, Dynamics Research Corporation: 1-54.

- Mitchell, D. K. (2000). Mental workload and ARL workload modeling tools. (ARL-TN-161) Aberdeen Proving Ground, MD: Army Research Laboratory.

- Mitchell, D., Samms, C., Henthorn, T., Wojciechowski, J. (2003). Trade Study: A Two-Versus Three-Soldier Crew for the Mounted Combat System (MCS) and Other Future Combat System Platforms. (ARL-TR-3026) Aberdeen Proving Ground, MD: Army Research Laboratory.

Thanks for Coming!